UNCLASSIFIED MASTER COPY - FOR REPRODUCTION PURPOSES (2)

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

DTIC FILE COPY

AD-A198 761

1a. REPORT SECURITY CLASSIFICATION: Unclassified	1b. RESTRICTIVE MARKINGS
DULE	3. DISTRIBUTION/AVAILABILITY OF REPORT: Approved for public release; distribution unlimited.
BER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S): ARO 22835.9-MA
6a. NAME OF PERFORMING ORGANIZATION: North Carolina State Univ. — 6b. OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION: U. S. Army Research Office
6c. ADDRESS (City, State, and ZIP Code): Raleigh, NC 27695-7913	7b. ADDRESS (City, State, and ZIP Code): P. O. Box 12211, Research Triangle Park, NC 27709-2211
8a. NAME OF FUNDING/SPONSORING ORGANIZATION: U. S. Army Research Office — 8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER: DAAL03-86-K-0039
8c. ADDRESS (City, State, and ZIP Code): P. O. Box 12211, Research Triangle Park, NC 27709-2211	10. SOURCE OF FUNDING NUMBERS: PROGRAM ELEMENT NO. / PROJECT NO. / TASK NO. / WORK UNIT ACCESSION NO.

11. TITLE (Include Security Classification)

Documentation of Bidnet: Project Bidding for CPM and PERT Activity Networks

12. PERSONAL AUTHOR(S)
Salah E. Elmaghraby and David J. Michael

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
Technical	FROM ____ TO ____	June 1988	28

16. SUPPLEMENTARY NOTATION
The view, opinions and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy, or decision, unless so designated by other documentation.

17. COSATI CODES		
FIELD	GROUP	SUB-GROUP

18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
Bidnet, Project Bidding, Activity Networks, Software, Logic, Program Logic, Probabilistic Durations, Confidence Levels. (SDW)

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

Software is presented (designed for the IBM PC/AT or compatibles, in ProBasic) and its underlying logic for the determination of the bid on a project given its activities, their durations, and their individual cost functions. Program logic admits deterministic as well as probabilistic durations and costs, and any desired confidence (in the probabilistic case) on the bid estimates.

DTIC ELECTE AUG 1 1 1988 S D H

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT: ☐ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION: Unclassified
22a. NAME OF RESPONSIBLE INDIVIDUAL	22b. TELEPHONE (Include Area Code) — 22c. OFFICE SYMBOL

DD FORM 1473, 84 MAR — 83 APR edition may be used until exhausted. All other editions are obsolete.

SECURITY CLASSIFICATION OF THIS PAGE
UNCLASSIFIED

ARO 22835.9-MA

OPERATIONS RESEARCH

NORTH CAROLINA STATE UNIVERSITY AT RALEIGH

88

DOCUMENTATION OF BIDNET: PROJECT BIDDING FOR CPM AND PERT ACTIVITY NETWORKS

by

Salah E. Elmaghraby
David J. Michael

OR REPORT NO. 221 JUNE 1988

This research was partially supported by ARO Contract No. DAAL03-86-K-0039, which is gratefully acknowledged.

DOCUMENTATION OF BIDNET: PROJECT BIDDING FOR CPM AND PERT ACTIVITY NETWORKS

by

Salah E. Elmaghraby and David J. Michael
North Carolina State University, Raleigh NC, 27695-7913

Abstract

We present software (designed for the IBM PC/AT or compatibles, in ProBasic) and its underlying logic for the determination of the bid on a project given its activities, their durations, and their individual cost functions. Program logic admits deterministic as well as probabilistic durations and costs, and any desired confidence (in the probabilistic case) on the bid estimates.

Introduction

The theoretical justification for the logic of this computer program may be found in [Elmaghraby (1988)]. This report is devoted to some explanatory remarks on the construction and operation of the program.

The project bidding program is composed of three separate programs: EDNET, BIDNET, and BIDTIME. Program EDNET allows you to either input the description of a new network or modify an existing one, and store the result as a specially formatted file. Program BIDNET reads the formatted file and, through a series of network reductions (simplifications), it calculates the time and cost distributions of each "keyevent" (KE), where "cost" is defined in terms of the activities *contributing to the realization of the* KE.. Program BIDTIME calculates the costs incurred as of the keyevent expected realization time, *irrespective of whether or not the activities did contribute to the realization of the* KE. There is another generic difference between the two bid-estimating programs: BIDNET is basically *probabilistic* in its calculations, while BIDTIME is basically *deterministic*. This is mainly due to the extreme difficulty (computer-wise) of calculating the "cost incurred to date" at the (expected) time of realization of a KE.

EDNET Description

Load EDNET if you are creating or modifying a network description: otherwise, if your network is already on file load BIDNET.

Run EDNET. Select DIRECTORY option to specify the directory in which you wish to create a new network (or from which you wish to read an existing network).

Select READ option to read an existing network.

Select EDIT option to either change an existing network or to create a new one. When creating a new network, first select the PARAMETER option to specify the number of nodes, arcs, and keyevents in the network, and indicate whether it is deterministic or probabilistic. If probabilistic also specify the desired maximum number of discrete intervals for the probability distributions. Also use the PARAMETER option to specify the interest rate which reflects the time value of money. This is used to compound the cost on each activity. Next select NETWORK option to describe each arc. Fill in the starting and ending nodes, the costs (costs are assumed to be linear w.r.t. time), and the durations. For probabilistic networks the duration will be a distribution which you specify from the menu provided as follows.

With the duration field highlighted, press enter and choose the desired distribution. For a DISCRETE distribution, the *probability mass points must be equally spaced.* Also use the DISCRETE distribution with only one mass point for deterministic arcs. For all other distributions a minimum and a maximum time value must be specified. Great care must be taken in specifying these values for the NORMAL, EXPONENTIAL, BETA, and GAMMA distributions because otherwise they may not be discretized correctly.

Return to MAIN and **select** MILESTONE.

The MILESTONE option allows each milestone (keyevent) to be identified by node number. It also requires the user to specify the desired confidence level for the realization time and the realization cost imputed as percentage (e.g., 80 to indicate 80% confidence).

BIDNET Description

Load BIDNET either *ab initio* (if the project network is on file) or as a sequel to EDNET.

Run BIDNET.

Select DIRECTORY option to specify the location of the project. **Select** READ option to choose the desired network file. Then **select** RUN option to execute the network. **Select** PRINTER option on the RUN menu (a toggle) only if you wish to send screen output to the printer.

There are three execution options under RUN: REDUCE, TIME, and BID.

QUALITY INSPECTED 2

For
Distribution/
Availability Codes
Dist | Avail and/or Special
A-1

1. REDUCE will only show the steps taken to reduce (simplify) the network to a smaller and smaller number of arcs and nodes, until the network consists of just the first and last nodes connected by a single arc. This is done using the LOGICAL sub-option. The DETAIL sub-option will additionally do the actual convolution (combining) of the cost-time distributions and print the resulting bivariate distribution for the new equivalent arc. This sub-option allows for checking the program's logic, i.e. debugging.
2. The TIME option is to be selected only if the times of realization of the KEs and nothing else are of interest, since the times of realization of KEs are re-computed under the BID option. TIME finds the subnetwork for each keyevent and reduces that subnetwork as in REDUCE to an equivalent single arc network. The marginal realization time distribution is displayed and possibly printed. Watch for the "Number of Nodes" decrease from N-1 to 2 before it prints the results.
3. BID executes first the TIME option and displays the marginal time distribution for each KE. It then re-executes the network reduction for each KE. However, the costs for each arc are first adjusted according to the expected realization times of the keyevent subnetworks it is a part of, and according to which keyevent subnetwork is currently to be reduced. The marginal realization cost and the bivariate time-cost distributions are displayed and possibly printed for each keyevent. The bottom line of the screen keeps track of calculations.

Execution Details

The network description is read into "permanent" arrays which do not change until a different network file is selected. Distributions such as UNIFORM, NORMAL, etc., are discretized and the resulting discrete distributions are also stored in these arrays. Execution copies these arrays into work arrays (which are slightly larger to accommodate the reduction process) and provide for the cost dimension in the probability.

Subroutine SUBNET

This routine determines the keyevent subnetwork(s) to which each arc belongs. It uses a BFS (breadth first search) which flags each arc and node as it is found to be in the subnetwork. However it must first take the network in the work arrays which describe each arc, and create a new data structure which is basically a series of pointers and pointer lists.

Arc no.	Start node	End node
1	1	2
2	1	3
3	2	4
4	1	4
5	3	5

Node:	1	2	3	4
P0%:	1	5	7	9

Index:	1	2	3	4	5	6	7	8	9
ARCND%:	1	2	4	0	3	0	5	0	0

Node:	1	2	3	4
LSTND%:	0	1	0	2

Figure 1. Sample network and its associated lists.

The array ARCND% stores the lists of arc numbers for each node for those arcs emanating from that node. Each sublist stores a zero to indicate the end of the sublist. The array P0% stores the index for the beginning of

each node's sublist in ARCND%. For example, given the following network of four nodes, ARCND% and PO% would be as shown in Fig. 1
The 0's in ARCND% separate the nodes in ascending order: arcs #1, 2, and 4 emanate from node 1, arc #3 emanates from node 2, arc #5 emanates from node 3, and no arc emanates from node #4. Consequently, the number of 0's appearing in ARCND% corresponds exactly to the number of nodes of the network. PO% lists the index number of ARCND% that corresponds to each node of the network. Thus, for instance, node #1 corresponds to index 1 in ARCND%, while node #3 corresponds to index 7 in ARCND%.

A BFS is done which simply follows along a path until 1) the keyevent itself is found; 2) a node already identified as being in the subnetwork is found; or 3) there are no more arcs to follow. Arcs found to be in the subnetwork are stored in the work arrays. The path currently being followed is recorded in the array LSTND%. This actually records for each node the index of the last node encountered on the path immediately preceding the current node. For example, if the path 1-2-4 were followed in the network above, LSTND% would be as shown in Fig. 1 (#1 has no precedor, #2 is preceded by #1, #3 has no precedor in this path, and #4 is preceded by #2).

Network Reduction

In order to reduce the (sub-)network a different data structure is required. The key component is a node-incidence matrix where the row index represents "into" nodes, the column index represents "out of" nodes, and the number stored is the arc number if an arc exists for the combination. This allows the network to be referenced by nodes instead of arcs.

The node-incidence matrix is used to determine how many arcs enter and leave each node. A node with either one arc entering or one arc leaving, or both, is selected and called VSTAR. The arc work-arrays have their empty slots chained together (using the starting node pointer NS%). END% points to the beginning of the chain. The slots for new arcs are taken from this list and the eliminated arcs are added to it. This method requires very little extra space in the work arrays to perform the reduction.

The Combination of Arc Distributions

In the (sub-)network reduction, two arcs are replaced by one equivalent arc. This requires that the two activity time-cost distributions be combined into one equivalent distribution.

In a *series* reduction the two time distributions must be added since the duration of the new arc is the sum of the two.

Parallel reduction multiplies the time distributions, since the longer duration determines the duration of the new arc.

The *cost* distributions, however, are always added as each of the activities being combined will be performed, whether in series or in parallel. so that both costs are incurred.

The first step in combining the distributions is to find the range of the distribution of the new arc . The time values stored for each arc are probability "mass" points representing the center of each discrete probability interval. These points must always be equally spaced so that the left-hand minimum and right-hand maximum for any time distribution can be calculated. See Fig. 2.

The range for an arc which results from a series reduction is found by adding the two minima and the two maxima of its constituent arcs. This range is then divided equally into either the desired maximum number of intervals (part of the network description in the file), or into the resulting number of possible time values, whichever is less. For example, if up to five intervals are desired but there are only two time values for each distribution, then the range is divided into only four(=2×2) intervals. See Example 1.

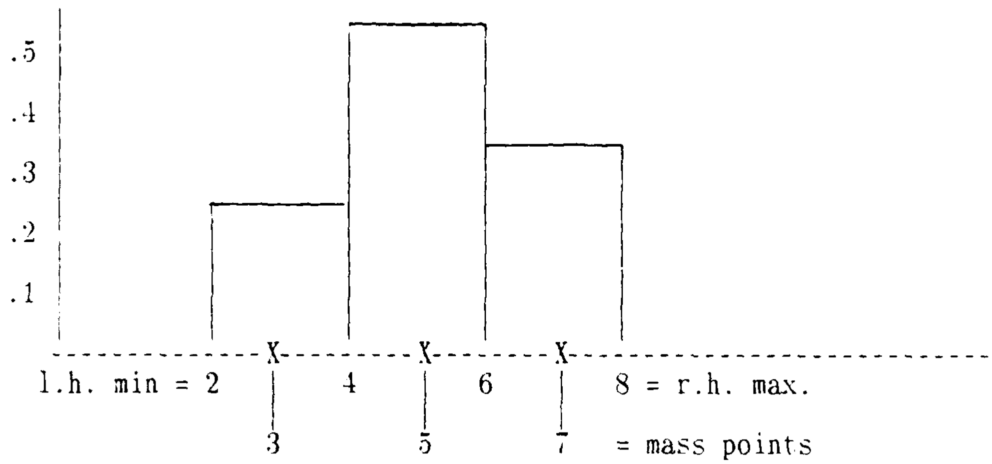

Figure 2. Time Dimension Divisions

Example 1. (Desired maximum number of intervals = 5)
Arc 1: Range = 20, 50
Mass points = 23, 29, 35, 41, 47
Arc 2: Range = 30, 60
Mass points = 35, 45, 55
Arc 3 = *series* reduction of arcs 1 & 2
Range = 50, 110
Mass points = 56, 68, 80, 92, 104

Arc 3 = *parallel* reduction of arcs 1 & 2
Range = 30, 60
Mass points = 35, 45, 55

The range for an arc duration resulting from a parallel reduction is somewhat more involved as there are several cases which must be considered. If both time distributions have constant duration (hence they are deterministic durations), the larger value is selected and effectively only the cost distributions are added. Otherwise the program must decide if one range dominates the other, which means that one of the distributions has both the larger maximum and the larger minimum. If so, then that range with its divisions is used. Otherwise, either one of the two ranges is contained in the other, or the two ranges overlap. In either case the larger minimum and the larger maximum of the two are used for the new arc with the desired maximum number of intervals; see example 2.

<u>Example 2</u>. (Desired maximum number of intervals = 5)
Arc 1: Range = 20, 50
Mass points = 23, 29, 35, 41, 47
Arc 2: Range = 30, 40
Mass points = 32, 34, 36, 38
Arc 3 = *parallel* reduction of arcs 1 & 2
Range = 30, 50
Mass points = 32, 36, 40, 44, 48

Once the range for the new time distribution has been determined, the right-hand value (not the mass point value) for each of the discrete intervals is stored. These will be changed later to probability mass points instead.

The actual convolution is done in three main steps. For each interval

Milestone number	Node number of milestone	Compl Time confidence	Compl Cost confidence

of the new distribution (as stored above), a feasible combination of times from the contributing distributions must first be identified. For *series* reduction this means that the sum of the two time values is in the interval if the sum is less than or equal to the interval's r.h. value while not also being in the previous interval (less than r.h. value of the last interval). For *parallel* reduction the individual time values must both be less than or equal to the interval's r.h. value while not also being less than the previous interval's r.h. value.

Once a feasible combination is identified, all of the possible costs (and their corresponding probabilities) for the time combination are stored in arrays as heaps. (Costs with zero probability are ignored.) Unlike the time dimension, it is not possible to determine the resulting range of costs for each time beforehand: there may be more than one combination of times to include. Therefore, the left-hand minimum and right-hand maximum costs for each time interval (which are stored in the work arrays) are respectively added together and compared to the current cost range. In this way the range for each time interval is eventually determined. Note that even though the cost intervals are always equally spaced, the minimum and maximum values must be stored since the corresponding time value would most likely represent a range of values itself. Otherwise, there will only be single cost values which do not correspond to constant time durations and there would be no way of calculating the l.h. minimum and r.h. maximum from only one cost value.

After all of the feasible time combinations have been identified for an interval and the program has determined the minimum and maximum costs for the interval, it will have a heap with all of the possible combined costs and their associated probabilities. The cost range can then be divided into the desired maximum number of intervals, or into the resulting number of cost values if that is less. The right-hand value of each interval is stored and the heap of costs is repeatedly accessed for those costs belonging in the interval. The cost interval values are changed to mass points before moving on to the next time interval. See Tables 1-3 (extracted from Tables 3, 4, and 5 in the companion report by [Elmaghraby. (1988)]).

@@ Tables 1, 2, 3 here @@

Marginal Time and Cost Distributions

The marginal time distribution and *expected* cost are easily calculated from the bivariate distribution by summing the probabilities for that interval. The marginal cost distribution is more difficult to calculate. However, the logic is very similar to that used for convolution of costs. All the cost values for the *equivalent* keyevent arc are stored as a heap and the range is determined from the stored l.h. minima and r.h. maxima. The costs are removed in order and the marginal distribution is printed as each interval is completed. The calculation of the marginal distributions is also used to identify the realization time and cost that meet the specified confidences for each for the keyevent.

BIDTIME Description

BIDTIME is similar to BIDNET except that it handles the activities' costs differently. This apparently innocuous difference introduces fundamental changes in the perspective of cost recoupment, and in the method of cost accounting. It also necessitated a separate program.

For instance, under BIDNET it was possible to secure pdf's for both time and cost; but this is no longer feasible under BIDTIME. The latter uses the *average realization time* of the keyevent to determine costs. And for another, under BIDNET the cost of an activity is *apportioned* among the KE's to which the activity contributes, but under BIDTIME the analysis is more complex because we must take into account activities that do not contribute the the KE under consideration.

Under BIDTIME, calculations are based on the expected times of realization of the KE's, which must be determined first. At the (expected) time of realization of the KE under consideration, which will be referred to as "now", the activities of the network fall into either one of two classes: (i) activities that contribute to the realization of this KE (and possibly also other KE's); (ii) activities that do not contribute to the realization of this KE. These two classes are treated differently as follows.

(i) The cost of these activities is completely recouped at the (expected) time of realization of the KE, assumed to be "now".

(ii) These activities are, in turn, divided into two categories: "ongoing" and "external". The *ongoing* activities are those with expected start time before "now" and expected completion time after "now". A proportion of the cost of an ongoing activity is accounted for either wholly or partially "now". The *external* activities are those with expected start time after "now". Their cost does not enter into account at the expected time of realization of the KE under consideration.

To clarify the "proportion" of the cost of an ongoing activity that is accounted for "now", we note that there are *two* proportionalities that come into play. The first is proportionality of expected encumbrances relative to the expected duration of the activity. That is, if an activity has expected duration of 50 days, and 22 days have elapsed since its start, it is supposed to have incurred 44% of its total cost. The second is proportionality concerned with the negotiated agreement between contractor and owner relative to cost recoupment. Prior agreement may allow the recoupment of the full 44% of the cost of the activity, or only a fraction α, $0 < \alpha < 1$, of the expected costs incurred to date. For instance, if $\alpha = .6$ then the contractor may account for only 26.4% (= .6×44%) of the cost of the activity "now". Of course, care must be taken that no "double accounting" takes place, and that when the activity is completed its cost is fully accounted for.

Sample Session

The following is a running account of the conduct of a session in which a project network (called Expl) is created under EDNET and processed under BIDNET as well as under BIDTIME. To render the procedure more accessible the narrative is illustrated with plates that mirror the display on the monitor screen and present a "hard copy" of the output when the latter is directed to the printer.

In the following description we use the following notation:

"load xxx"	= F3, type xxx;
"run xxx"	= F2;
"select xxx"	= highlight "xxx" on the menu and hit "Return";
"enter xxx"	= type the desired information then hit "Return" if required.

EDNET:

At the prompt of A>, **enter** PB to load the Probasic language. **Load** EDNET. **Run** EDNET. When compilation is complete the MAIN MENU will appear on the screen, Plate 1. **Select** DIRECTORY where you wish the network to be saved (or from which the network is to be read). In this example, we selected drive A: Plate 2. Proceeding to introduce a new project network, select EDIT option. From the EDIT menu **select** PARAMETER to specify the network parameters. **Select** QUIT to return to EDIT menu. **Select** NETWORK to enter the required arcs' data. Plate 3. In this example, under network TYPE select "probabilistic". Under column "Distribution" **hit** "Return" to invoke the menu of available pdf's, as exhibited in Plate 4. For instance, for arc 3 we have **highlighted** UNIFORM, which now invokes you to **enter** the minimum (=

0) and maximum (= 5) durations, as shown in Plate 5. When finished with entering the parameters of the network **select** QUIT to return to the EDIT menu. Now **select** MILESTONE to specify the keyevent(s), Plate 6. At this point, you may **select** PRINT if a hard copy of the specifications of the network is desired. **Select** QUIT to return to the main menu and **select** SAVE. You shall be prompted to give the project network a name. In this example the network was given the name "Expl", to which the program automatically added the ".DAT" for "data", Plate 7. **Select** EXIT to leave EDNET. The editing of the network (in this case the insertion of a new network) is complete.

In general one loads either BIDNET or BIDTIME according to the application. We shall illustrate the operation of both.

BIDNET

Load BIDNET and **run** it. The Main Menu will appear. **Select** DIRECTORY to specify the drive in which the network was saved (in this case a:\), then **select** READ to load the network created by EDNET (in this case, one reads Expl.DAT). **Select** VIEW to display the network for verification. The full description of the network is displayed as shown in Plate 9. Note that one *cannot edit* the network at this point of the process: the VIEW facility is just to "view" the network before expending time on its analysis. Any desired changes must be performed in the EDNET program, which, at this point, means that you would exit BIDNET (quite a heavy penalty to pay for mistakes made in EDNET!). **Verify** the output destination, and change to "PRINTER" if desired (which is what we opted for). Note the remark on the screen "Printer flag (currently on)". **Select** QUIT to return to the Main Menu. **Select** RUN and three options shall be displayed: **select** either TIME, or REDUCE, or BID. Plate 9 shows the output from TIME; Plate 10 shows the output from the BID option.

BIDTIME

To load and run BIDTIME, select DIRECTORY, READ, and VIEW following similar steps to BIDNET. We have opted for "output to screen" to be able to print individual screens. When BIDTIME is run, respond to the prompt on the "proportion of costs paid for ongoing activity" by 0.5, Plate 12, which means that 50% of the cost incurred "to date" of an ongoing activity may be recovered. Finally, we **selected** PRINTER to secure the full output of this program on the printer, Plates 13 and 14.

Reference

Elmaghraby, S.E. (1988). "Project bidding under deterministic and probabilistic activity durations", OR Report No. 220, NC State University, Raleigh NC, 27695-7913.

Table 1
Time and Cost Distributions
(Desired maximum number of points = 5)

	Time	Prob	Cond'l Cost	Cond'l Prob.	Range Duratn	Range Cost
Arc 1:					[20,50]	
	25	.05	105	1.0		[95,115]
	35	.50	120	0.20		[115,145]
			130	0.30		
			140	0.50		
	45	.45	135	4/9		[130.150]
			140	1/3		
			145	2/9		
Arc 2:					[40,60]	
	45	.35	50	4/7		[45,65]
			60	3/7		
	55	.65	65	3/13		[60,80]
			70	6/13		
			75	4/13		

Table 2
Derived Arc 3, Series Reduction

Duratn	Prob.	Duratn of Act. 1	2	Range	Cost Value	Cond'l Prob.	Abs. Prob.
70	.0175	25	45	[140,180]	155	4/7	.0100
					165	3/7	.0075
80	.0325	25	55	[155,195]	170	3/13	.0075
					175	6/13	.0150
					180	4/13	.0100
80	.175	35	45	[160,210]	170	1.2/7	.030
					180	2.9/7	.0725
					190	2.3/7	.0575
					200	.6/7	.0150
		Summary of Cost Values @ Duration 80					
80	.2075			[155,210]	170	.1807	.0375
					175	.0723	.0150
					180	.3976	.0825
					190	.2771	.0575
					200	.0723	.0150
90	.3250	35	55	[175,225]	185	.9/13	.0225
					190	1.8/13	.0450
					195	2.7/13	.0675
					200	3/13	.0750
					205	2.6/13	.0650
					210	1.2/13	.0300
					215	.8/13	.02
90	.1575	45	45	[175,215]	185	16/63	.04
					190	12/63	.03
					195	20/63	.05
					200	1/7	.0225
					205	6/63	.0150
		Summary of Cost Values @ Duration 90*					
90	.4825			[175,225]	180	.064767	.03125
					190	.341969	.165
					200	.406736	.19625
					210	.165803	.080
					220	.020725	.010
100	.2925	45	55	[190,230]	200	12/117	.03
					205	53/117	.0825
					210	24/117	.06
					215	40/117	.10
					220	8/117	.02

*Note that we had to collapse 7 points (185,190,195,200,205,210,215) into 5 points in conformity with the restriction imposed on the number of points of any variable (time or cost).

Table 3
Derived Arc 3, Parallel Reduction

Duratn	Prob.	Range	Cost Value	Cond'l Prob.	Abs. Prob.
45	0.35	[140,230]	155	.02857	
			165	.02143	
			170	.05714	
			180	.12857	
			185	.114286	
			190	.29286	
			195	.142857	
			200	.17143	
			205	.042857	
		Summary of Cost Values @ Duration 45*			
45	0.35	[140,230]	149	.021427777	.0075
			167	.111902775	.03916
			185	.520239111	.18208
			203	.341668444	.11958
			221	.004761888	.00167
55	0.65	[155,230]	170	.011538461	
			175	.023076923	
			180	.015384615	
			185	.023769230	
			190	.046153846	
			195	.065384615	
			200	.115384615	
			205	.230769230	
			210	.269230769	
			215	.169230768	
			220	.030769230	
		Summary of Cost Values @ Duration 55*			
55	0.65	[155,230]	162.5	.009615	.006250
			177.5	.057397	.037308
			192.5	.203551	.132308
			207.5	.575000	.373750
			222.5	.155127	.100833

*Note that we had to collapse 7 points (185.190.195.200,205,210,215) into 5 points in conformity with the restriction imposed on the number of points of any variable (time or cost).

```
MAIN-MENU:   EDIT   DIRECTORY   READ   SAVE   PRINT   EXIT

     EDIT     : edit current network
     DIRECTORY: select active directory, e.g. a:\
     READ     : select a data file from the present directory, e.g. PROBA
     SAVE     : save current data on a file
     PRINT    : print current data on the printer
     EXIT     : exit to PB




     Change the directory
```

Plate 1

```
+------------------------------------------+
|                                          |
|  Enter directory  (a:\) :?a:\            |
|                                          |
+------------------------------------------+
```

Plate 2

Arc Number	Starting node	Ending node	Fixed cost	Variable cost	Distribution	
1	10	20	11.00	1.25	TRIANGULAR	(0, 5, 10)
2	10	30	11.50	1.35	EXPONENTIAL	(1)
3	20	30	10.80	1.55	UNIFORM	(0, 5)
4	20	40	8.25	1.40	DISCRETE	(2)
5	30	40	9.50	0.95	NORMAL	(5, 1)

Press <space bar> to proceed

Plate 3

Arc Number	Starting node	Ending node	Fixed cost	Variable cost	Distribution	
1	10	20	11.00	1.25	TRIANGULAR	(0, 5, 10)
2	10	30	11.50	1.35	EXPONENTIAL	(1)
3	20	30	10.80	1.55	UNIFORM	(0, 5)
4	20	40	8.25	1.40	DISCRETE	(2)
5	30	40	9.50	0.95	NORMAL	(5, 1)

DISCRETE
UNIFORM
TRIANGULAR
NORMAL
EXPONENTIAL
GAMMA
BETA
QUIT

Press <space bar> to proceed

Plate 4

```
Enter minimum duration (Vmin) for activity  3 ( 0 ) ?
```

```
Enter maximum duration (Vmax) for activity  3 ( 5 ) ?
```

Plate 5

Milestone number	Node number of milestone	Compl Time confidence	Compl Cost confidence
1	40	80.00	90.00

Press <space bar> to proceed

Plate 6

```
A:\*.dat
EXPL.DAT                  357  <A>       AN11.DAT              1,246  <A>
EXPL3.DAT                 357  <A>
      3 entries.           1,960 bytes total.       29,696 bytes free.
```

```
Enter name of file (expl.DAT) : "expl    .DAT
```

Plate 7

Network Description
==================

Probabilistic network a:\expl.dat

Annual interest rate = 0.00027
Number of nodes = 4 nodes
Number of arcs = 5 arcs
Number of durations = 5 durations

Arc Number	Starting node	Ending node	Fixed cost	Variable cost	Duration	Probability	Range
1	1	2	11.0	1.25	TRIANGULAR	(0 , 5 , 10)	
2	1	3	11.5	1.35	EXPONENTIAL	(1)	0 , 4
3	2	3	10.8	1.55	UNIFORM	(0 , 5)	
4	2	4	8.2	1.40	0.00 10.00	0.400 0.600	
5	3	4	9.5	0.95	NORMAL	(5 , 1)	0 , 10

Press any key to proceed...

Node number of milestone	Comp Time confidence	Comp Cost confidence
4	80.0	90.0

Press any key to proceed...

Plate 8

Probability distribution function for keyevent 1

Completion time	Probability
2.50	0.0036
7.50	0.2538
12.50	1.4569
17.50	1.3998
22.50	0.0935

Expected completion time = 46.72

Plate 9

Cost density function for keyevent 1

Cost	Density
57.45	0.2434
70.24	0.9778
83.03	1.8529
95.83	0.1335

Key Event	Compl Time Confidence	Realization Time	Compl Cost Confidence	Realization Cost
1	80.0%	9.4	90.0%	-1227.1

Bivariate Time-Cost pdf for keyevent 1

Time	Cost	Pr(Cost & Time)
2.5		
	55.67	0.001
	64.90	0.002
	74.13	0.000
	83.37	0.000
	92.60	0.000
7.5		
	57.45	0.082
	70.24	0.147
	83.03	0.024
	95.83	0.000
	108.62	0.000
12.5		
	57.45	0.160
	70.24	0.715
	83.03	0.525
	95.83	0.056
	108.62	0.000
17.5		
	65.68	0.108
	76.64	0.621
	87.61	0.640
	98.57	0.031
	109.54	0.000
22.5		
	70.05	0.005
	80.04	0.042
	90.03	0.044
	100.03	0.002
	110.02	0.000

Plate 10

Keyevent node = 40

Completion time prob. dist. for keyevent 1

Compl. time	Prob.	Cumm prob.	Range = [0 , 25]
2.50	0.0023	0.0023	
7.50	0.0892	0.0915	
12.50	0.4260	0.5175	
17.50	0.4190	0.9365	
22.50	0.0534	0.9900	

Expected completion time = 14.54

Key event	Confidence of certainty	Realization time	Expected Real Time
1	80.00 %	18.37	14.54

Press any key to proceed...

Plate 11

Enter proportion of costs paid for on-going activity?.5

Enter -1 to quit

Plate 12

Keyevent realization costs according to realization TIMEs
Proportion of on-going activity cost to be paid = .5

Key Event	Compl Time Confidence	Realization Time	Compl Cost Confidence	Realization Cost
1	100.0%	14.5	100.0%	75.9

Press any key to proceed...

Plate 13

Keyevent node = 40

Completion time prob. dist. for keyevent 1

Compl. time	Prob.	Cumm prob.	Range = [0 , 25]
2.50	0.0023	0.0023	
7.50	0.0892	0.0915	
12.50	0.4260	0.5175	
17.50	0.4190	0.9365	
22.50	0.0534	0.9900	

Expected completion time = 14.54

Key event	Confidence of certainty	Realization time	Expected Real Time
1	80.00 %	18.37	14.54

Keyevent realization costs according to realization TIMEs
Proportion of on-going activity cost to be paid = .5

Key Event	Compl Time Confidence	Realization Time	Compl Cost Confidence	Realization Cost
1	100.0%	14.5	100.0%	75.9

Plate 14

ATE

LMED

8